Amendments to the Claims:

This listing of claims replaces all prior listings of claims:

Listing of Claims

1.   (Currently Amended) A method of synchronizing data objects between a first platform and a second platform comprising:

creating a set of generic messages identifying changes to the data objects since a previous synchronization;

converting the generic messages to adapted messages;

sending the adapted messages from the first platform to the second platform;

converting the adapted messages to generic messages on the second platform; and

updating the data objects on the second platform using the generic messages <u>converted from the adapted messages, the data objects on the second platform being updated based on stored mapping data, the stored mapping data being generated via user-generated input manipulating a mapping chart illustrating a mapping of variables from data objects on the first platform to data objects on the second platform.</u>

2.   (Original)   The method of claim 1, wherein creating the set of generic messages includes:

fetching the data objects from an application in the first platform; and

comparing the fetched data objects in the first platform with a replica of the data objects in the second platform to identify changes.

3.   (Original)   The method of claim 2, wherein fetching the data objects from an application in the first platform includes:

selecting data objects to be fetched; and

fetching only the selected data objects.

4.   (Currently Amended) The method of claim 2, further including categorizing the data objects <u>associated with the first platform</u> into a first category and a second category.

5.   (Original)   The method of claim 4, further including wherein creating the generic messages includes:

generating generic messages for only the first category of data objects.

6.   (Currently Amended) The method of claim 2, further including:

grouping <u>the</u> data objects <u>associated with the first platform</u> into a first transaction group and a second transaction group.

7.   (Currently Amended) The method of claim 6, further including:

sending a failure notification from the second platform to the first platform if the update <u>of</u> <u>one of the data objects</u> ~~on the data object~~ of the first transaction group <u>on</u> ~~in~~ the second platform fails; and

rolling back all updating of data objects of the first transaction group on the second platform upon the failure notification.

8.   (Original)   The method of claim 1, wherein converting the generic messages to adapted messages includes converting the generic messages to adapted messages based on the requirements of an underlying synchronization software.

9.　　(Original)　　The method of claim 1, wherein sending the adapted messages from the first platform to the second platform includes sending the adapted messages using an underlying synchronization software.

10.　　(Original)　　The method of claim 1, wherein updating the data objects on the second platform using the generic messages includes executing the generic messages so that they act on the data objects of the second platform.

11.　　(Currently Amended) A system for synchronizing data objects between a first platform and a second platform comprising:

a memory; and

a microprocessor coupled to the memory and programmed to execute at least one computer program to:

create a set of generic messages identifying changes to the data objects since a previous synchronization, the generic messages having a first format;

convert the generic messages to adapted messages, the adapted messages having a second format different than the first format;

send the adapted messages from the first platform to the second platform;

convert the adapted messages to generic messages on the second platform; and

update the data objects on the second platform using the generic messages converted from the adapted messages, the data objects on the second platform being updated based on stored mapping data, the stored mapping data being generated via user-generated input manipulating a mapping chart illustrating a

mapping of variables from data objects on the first platform to data objects on the
second platform.

12.   (Original)   The system of claim 11, wherein the microprocessor is further
programmed to:

fetch the data objects from an application in the first platform;

compare the fetched data objects in the first platform with a replica
of the data objects in the second platform to identify changes.

13.   (Original)   The system of claim 12, wherein the microprocessor is further
programmed to:

select data objects to be fetched; and

fetch only the selected data objects.

14.   (Original)   The system of claim 12, wherein the microprocessor is further
programmed to categorize the data objects into a first category and a second category.

15.   (Original)   The system of claim 14, wherein the microprocessor is further
programmed to only generate generic messages for the first category of data objects.

16.   (Original)   The system of claim 12, wherein the microprocessor is further
programmed to:

group data objects into a first transaction group and a second transaction group.

17.   (Original)   The system of claim 16, wherein the microprocessor is further
programmed to:

send a failure notification from the second platform to the first platform if the update on

the data object of the first transaction group in the second platform fails; and

roll back all updating of data objects of the first transaction group on the second platform

upon the failure notification.

18.   (Original)   The system of claim 11, wherein the microprocessor is further

programmed to convert the generic messages to adapted messages based on the requirements of

an underlying synchronization software.

19.   (Original)   The system of claim 11, wherein the microprocessor is further

programmed to send the adapted messages using an underlying synchronization software.

20.   (Currently Amended) A system for synchronizing data objects for a user between

a primary first platform and a plurality of auxiliary second platforms comprising:

a memory; and

processing means, coupled to the memory, to execute at least one computer program for:

creating a set of generic messages identifying changes to the data objects on the

primary platform since a previous synchronization;

accessing a database to obtain a user identifier, the user identifier being

associated with the user and linking to one or more device identifiers, the device

identifiers identifying the plurality of auxiliary platforms;

converting the generic messages to adapted messages for each of the auxiliary

platforms based on the obtained user identifier and the linked device identifiers;

sending the adapted messages from the primary first platform to the

corresponding auxiliary second platforms;

converting the adapted messages to generic messages on ~~each of~~ the auxiliary ~~second~~ platforms; and

updating the data objects on the respective auxiliary ~~second~~ platforms using the generic messages converted from the adapted messages.